Ex. No: 04	**JOINS AND SET OPERATIONS**
Date	

Objective:

To execute the given queries using set operators and joins.

Description:

Set Operators

The three *set operators* union, intersect and minus allow to serially combine more than one select statements. Although more than one select statement will then be present, only *one* result set is then returned. The following list briefly describes the three set operations supported by Oracle SQL:

1) UNION

union all is very similar to union, however, it dismisses duplicate rows found across different select statements:

select col_1, col_2, col_3 from table_1 union

select col_1, col_2, col_3 from table_2;

2) INTERSECT

intersect only returns the rows that are found in all select statements:

select col_1, col_2, col_3 from table_1 intersect

select col_1, col_2, col_3 from table_2;

3) MINUS

minus returns all rows from the first select statements except those who are duplicated in a following select statement:

select col_1, col_2, col_3 from table_1 minus

select col_1, col_2, col_3 from table_2;

SQL JOIN

The JOIN keyword is used in an SQL statement to query data from two or more tables, based on a relationship between certain columns in these tables.

Tables in a database are often related to each other with keys.

Different SQL JOINs

Before we continue with examples, we will list the types of JOIN you can use, and the differences between them.

JOIN: Return rows when there is at least one match in both tables

LEFT JOIN: Return all rows from the left table, even if there are no matches in the right table

RIGHT JOIN: Return all rows from the right table, even if there are no matches in the left table

FULL JOIN: Return rows when there is a match in one of the tables

SQL INNER JOIN Keyword

The INNER JOIN keyword return rows when there is at least one match in both tables.

Syntax

```
SELEC column_name(s)
FROM table_name1
INNER JOIN table_name2
ON table_name1.column_name=table_name2.column_name
```

PS: INNER JOIN is the same as JOIN.

SQL LEFT JOIN Keyword

The LEFT JOIN keyword returns all rows from the left table (table_name1), even if there are no matches in the right table (table_name2).

Syntax

```
SELECT column_name(s)
FROM table_name1
LEFT OUTER JOIN table_name2
ON table_name1.column_name=table_name2.column_name
```

SQL RIGHT JOIN Keyword

The RIGHT JOIN keyword Return all rows from the right table (table_name2), even if there are no matches in the left table (table_name1).

Syntax

```
SELECT column_name(s)
FROM table_name1
RIGHT OUTER JOIN table_name2
ON table_name1.column_name=table_name2.column_name
```

SQL FULL JOIN Keyword

The FULL JOIN keyword return rows when there is a match in one of the tables.

Syntax

```
SELECT column_name(s)
FROM table_name1
```

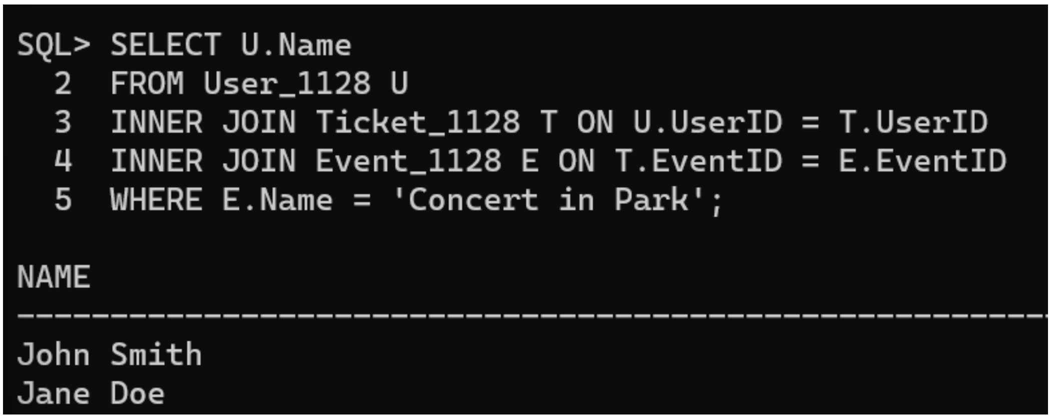
FULL OUTER JOIN table_name2

ON table_name1.column_name=table_name2.column_name

The JOIN keyword is used in an SQL statement to query data from two or more tables, based on a relationship between certain columns in these tables. Whenever a query is written which refers more than one table that needs the help of joins.

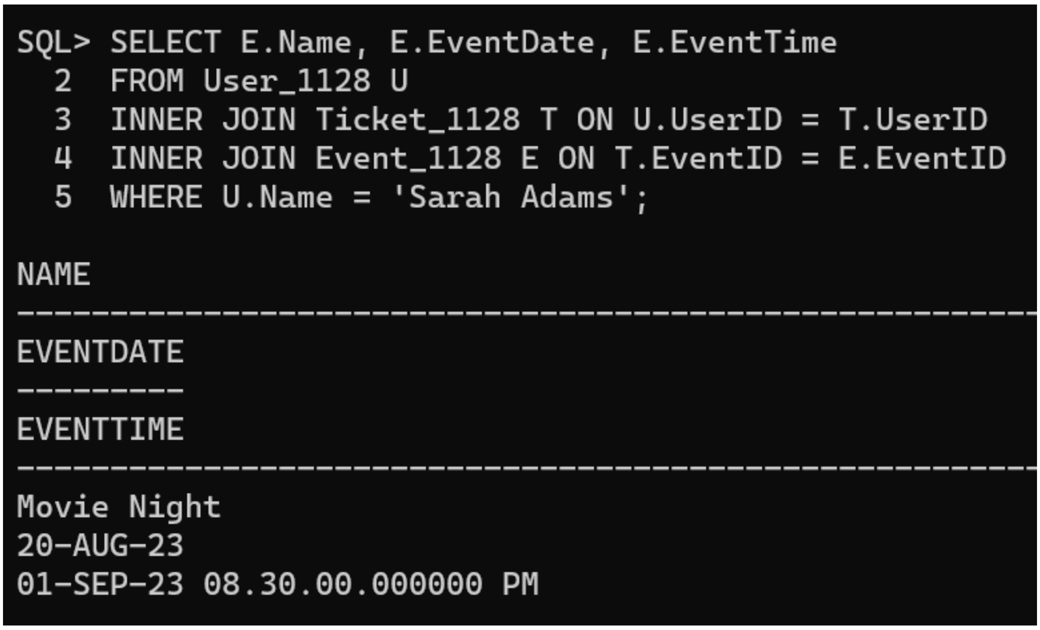
Questions:

1. Retrieve the names of users who have registered for the "Concert in Park" event:



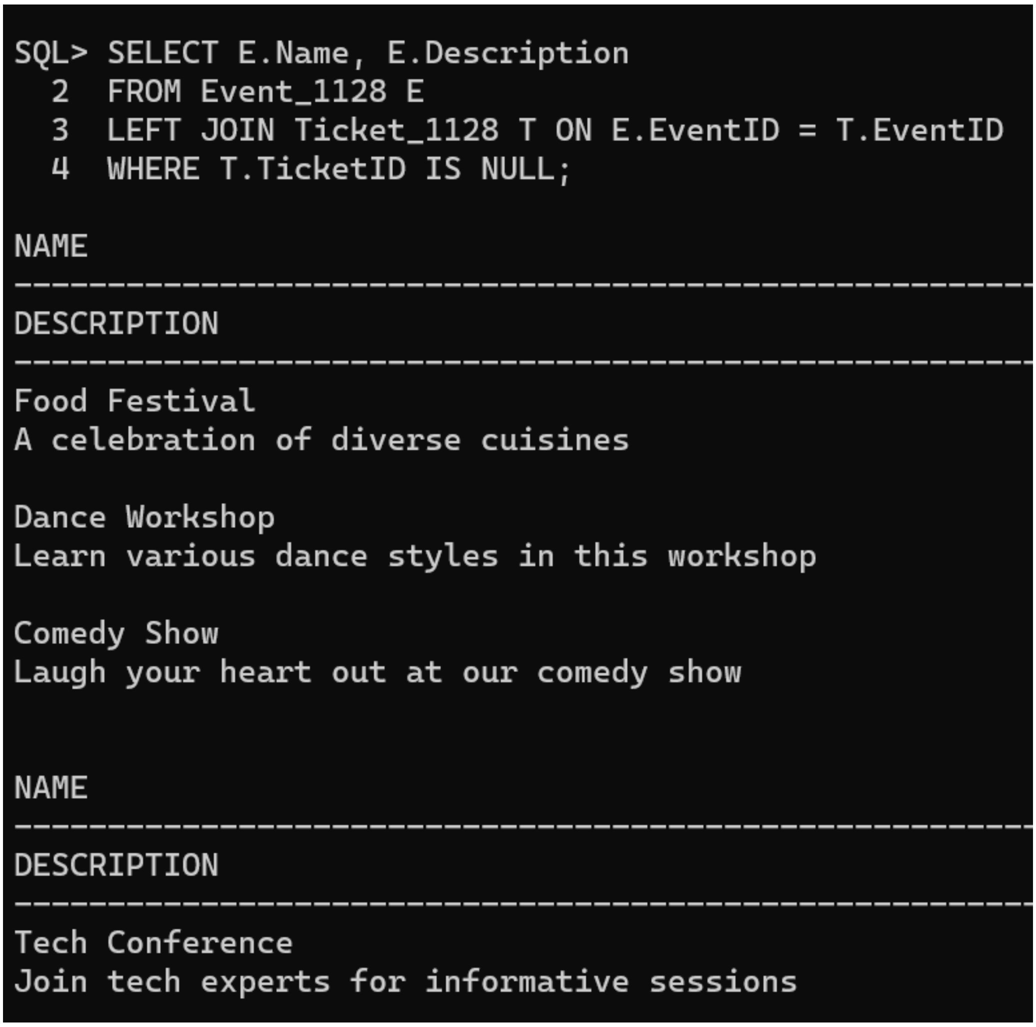
```
SQL> SELECT U.Name
     FROM User_1128 U
     INNER JOIN Ticket_1128 T ON U.UserID = T.UserID
     INNER JOIN Event_1128 E ON T.EventID = E.EventID
     WHERE E.Name = 'Concert in Park';

NAME
--------------------------------------------------------
John Smith
Jane Doe
```

2. Find the details of events (name, date, and time) that Sarah Adams has registered for.



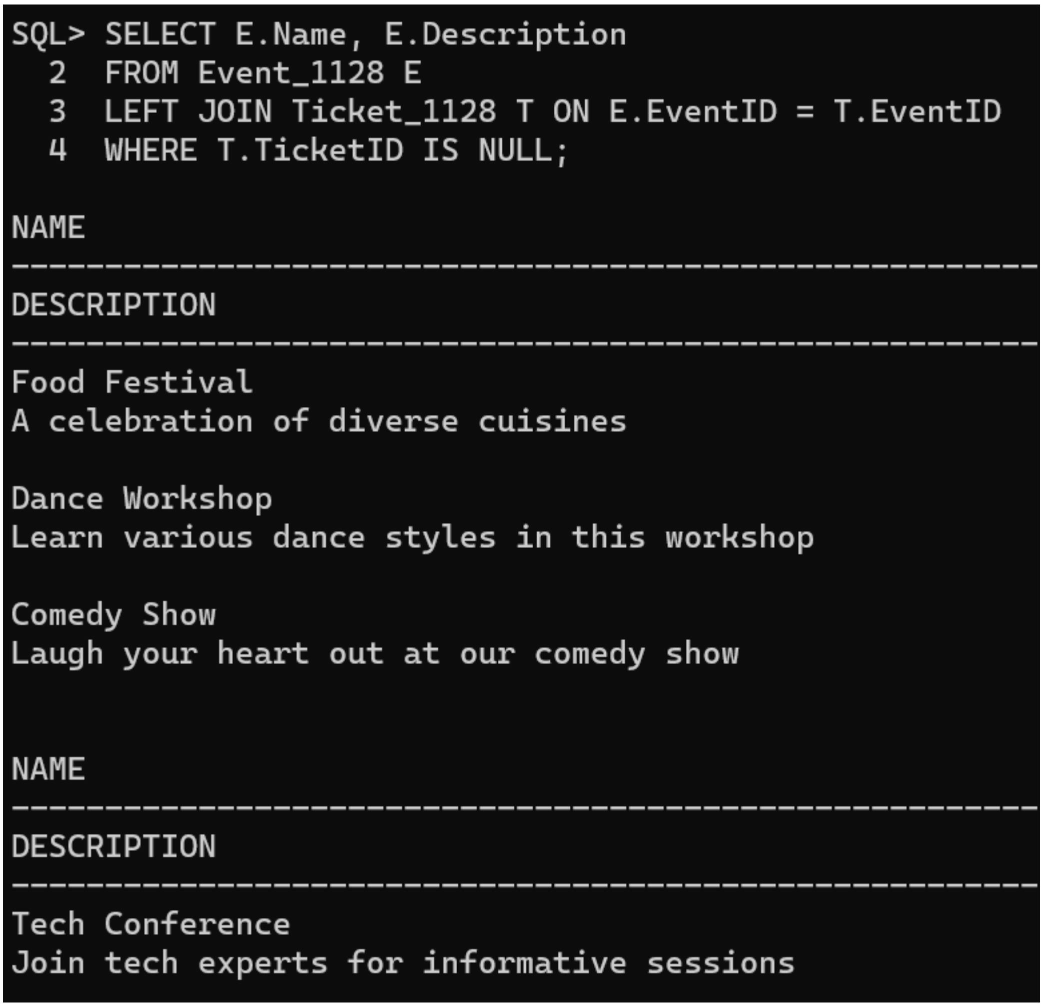
```
SQL> SELECT E.Name, E.EventDate, E.EventTime
     FROM User_1128 U
     INNER JOIN Ticket_1128 T ON U.UserID = T.UserID
     INNER JOIN Event_1128 E ON T.EventID = E.EventID
     WHERE U.Name = 'Sarah Adams';

NAME
--------------------------------------------------------
EVENTDATE
---------
EVENTTIME
--------------------------------------------------------
Movie Night
20-AUG-23
01-SEP-23 08.30.00.000000 PM
```

3. List the events (name and description) that do not have any registered participants.



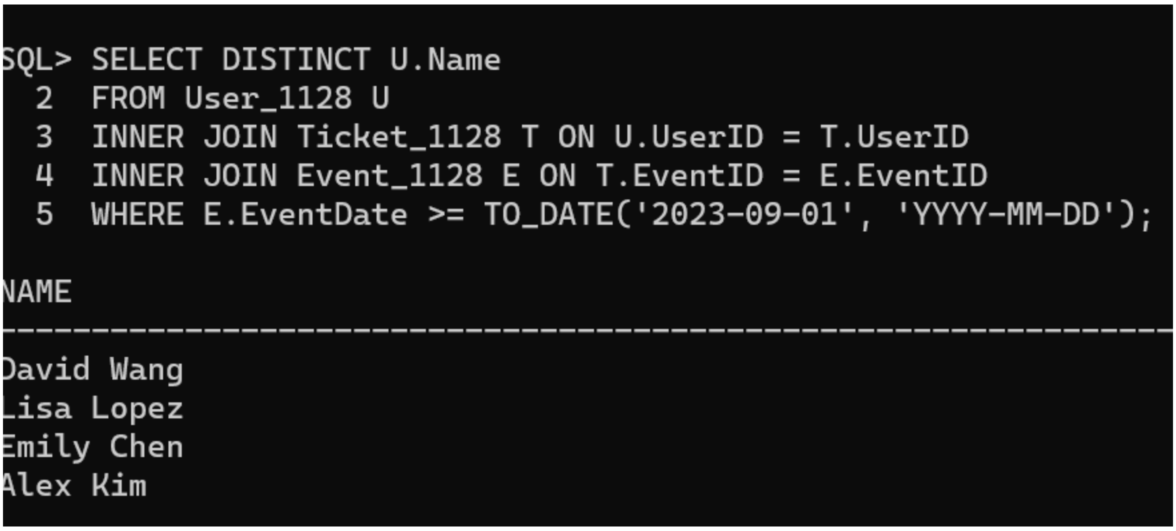
```
SQL> SELECT E.Name, E.Description
  2  FROM Event_1128 E
  3  LEFT JOIN Ticket_1128 T ON E.EventID = T.EventID
  4  WHERE T.TicketID IS NULL;

NAME
-------------------------------------------------------
DESCRIPTION
-------------------------------------------------------
Food Festival
A celebration of diverse cuisines

Dance Workshop
Learn various dance styles in this workshop

Comedy Show
Laugh your heart out at our comedy show


NAME
-------------------------------------------------------
DESCRIPTION
-------------------------------------------------------
Tech Conference
Join tech experts for informative sessions
```

4. Retrieve the names of users and the events they have registered for, along with the event dates.



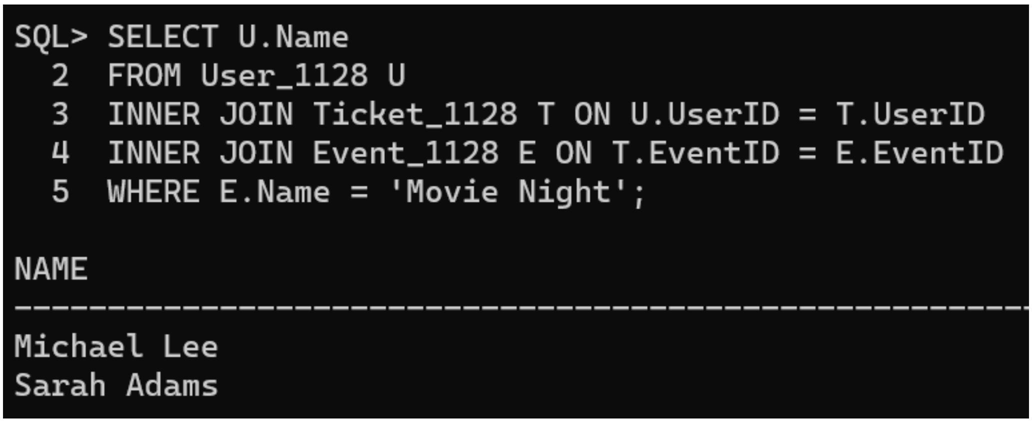
```
SQL> SELECT E.Name, E.Description
  2  FROM Event_1128 E
  3  LEFT JOIN Ticket_1128 T ON E.EventID = T.EventID
  4  WHERE T.TicketID IS NULL;

NAME
-------------------------------------------------------
DESCRIPTION
-------------------------------------------------------
Food Festival
A celebration of diverse cuisines

Dance Workshop
Learn various dance styles in this workshop

Comedy Show
Laugh your heart out at our comedy show


NAME
-------------------------------------------------------
DESCRIPTION
-------------------------------------------------------
Tech Conference
Join tech experts for informative sessions
```

5. Find the names of users who have registered for events taking place on or after September 1, 2023.



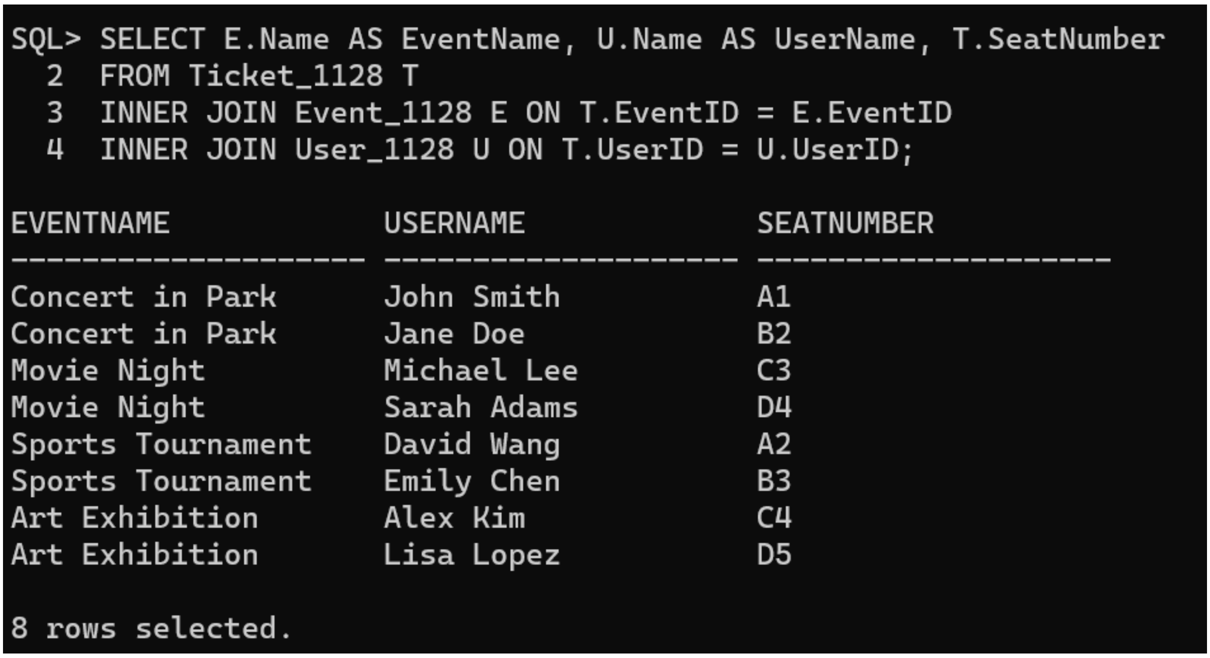
```
SQL> SELECT DISTINCT U.Name
  2  FROM User_1128 U
  3  INNER JOIN Ticket_1128 T ON U.UserID = T.UserID
  4  INNER JOIN Event_1128 E ON T.EventID = E.EventID
  5  WHERE E.EventDate >= TO_DATE('2023-09-01', 'YYYY-MM-DD');

NAME
--------------------------------------------------------------
David Wang
Lisa Lopez
Emily Chen
Alex Kim
```

6. Retrieve the names of users who have booked tickets for the "Movie Night" event.



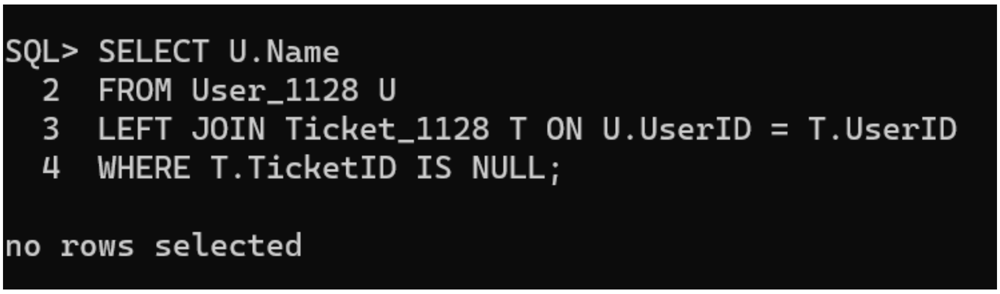
```
SQL> SELECT U.Name
     FROM User_1128 U
     INNER JOIN Ticket_1128 T ON U.UserID = T.UserID
     INNER JOIN Event_1128 E ON T.EventID = E.EventID
     WHERE E.Name = 'Movie Night';

NAME
--------------------------------------------------
Michael Lee
Sarah Adams
```

7. List the event names, user names, and seat numbers for all booked tickets.



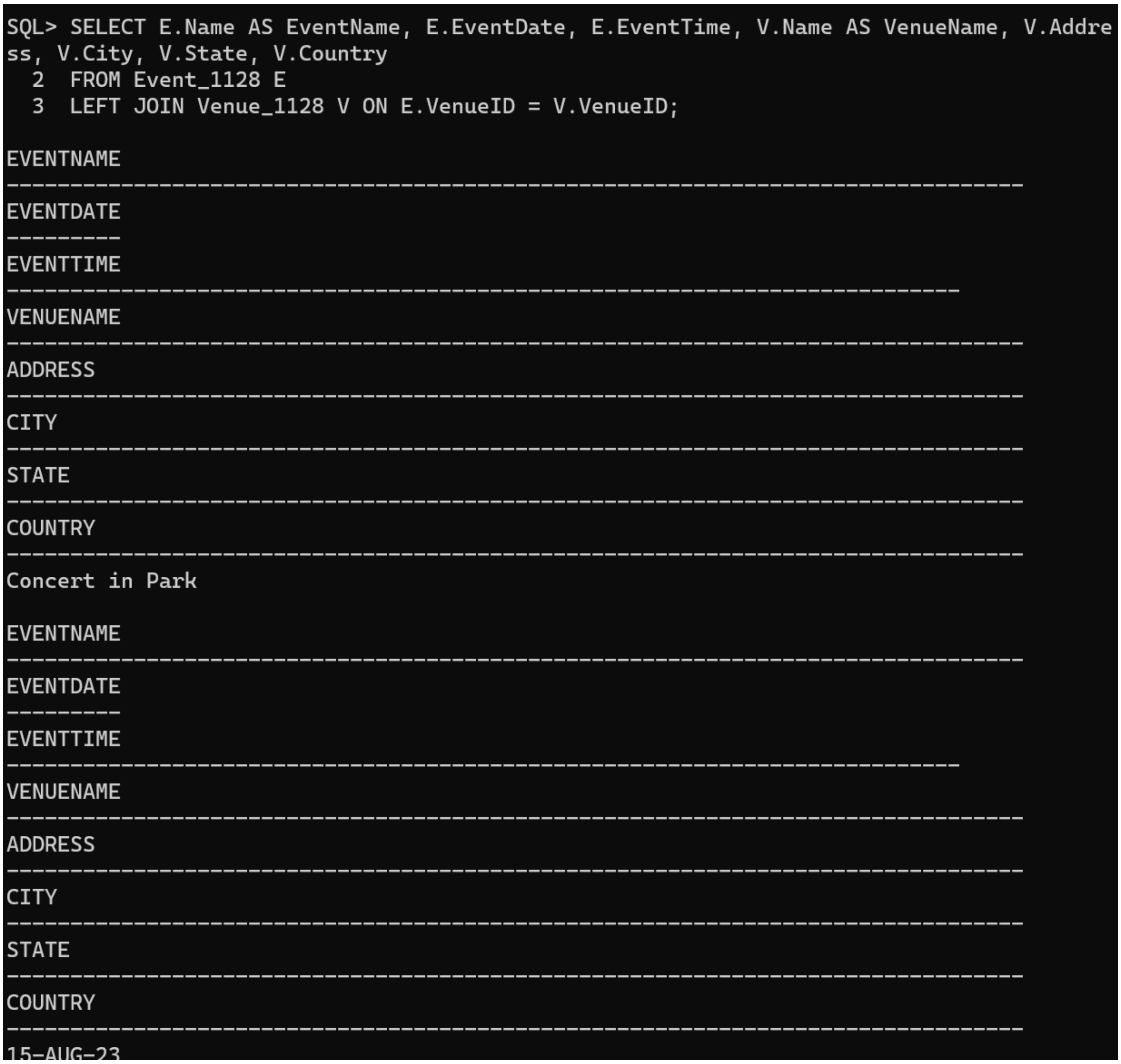
```
SQL> SELECT E.Name AS EventName, U.Name AS UserName, T.SeatNumber
     FROM Ticket_1128 T
     INNER JOIN Event_1128 E ON T.EventID = E.EventID
     INNER JOIN User_1128 U ON T.UserID = U.UserID;

EVENTNAME            USERNAME             SEATNUMBER
-------------------- -------------------- --------------------
Concert in Park      John Smith           A1
Concert in Park      Jane Doe             B2
Movie Night          Michael Lee          C3
Movie Night          Sarah Adams          D4
Sports Tournament    David Wang           A2
Sports Tournament    Emily Chen           B3
Art Exhibition       Alex Kim             C4
Art Exhibition       Lisa Lopez           D5

8 rows selected.
```

8. Find the names of users who have not booked any tickets for any event.



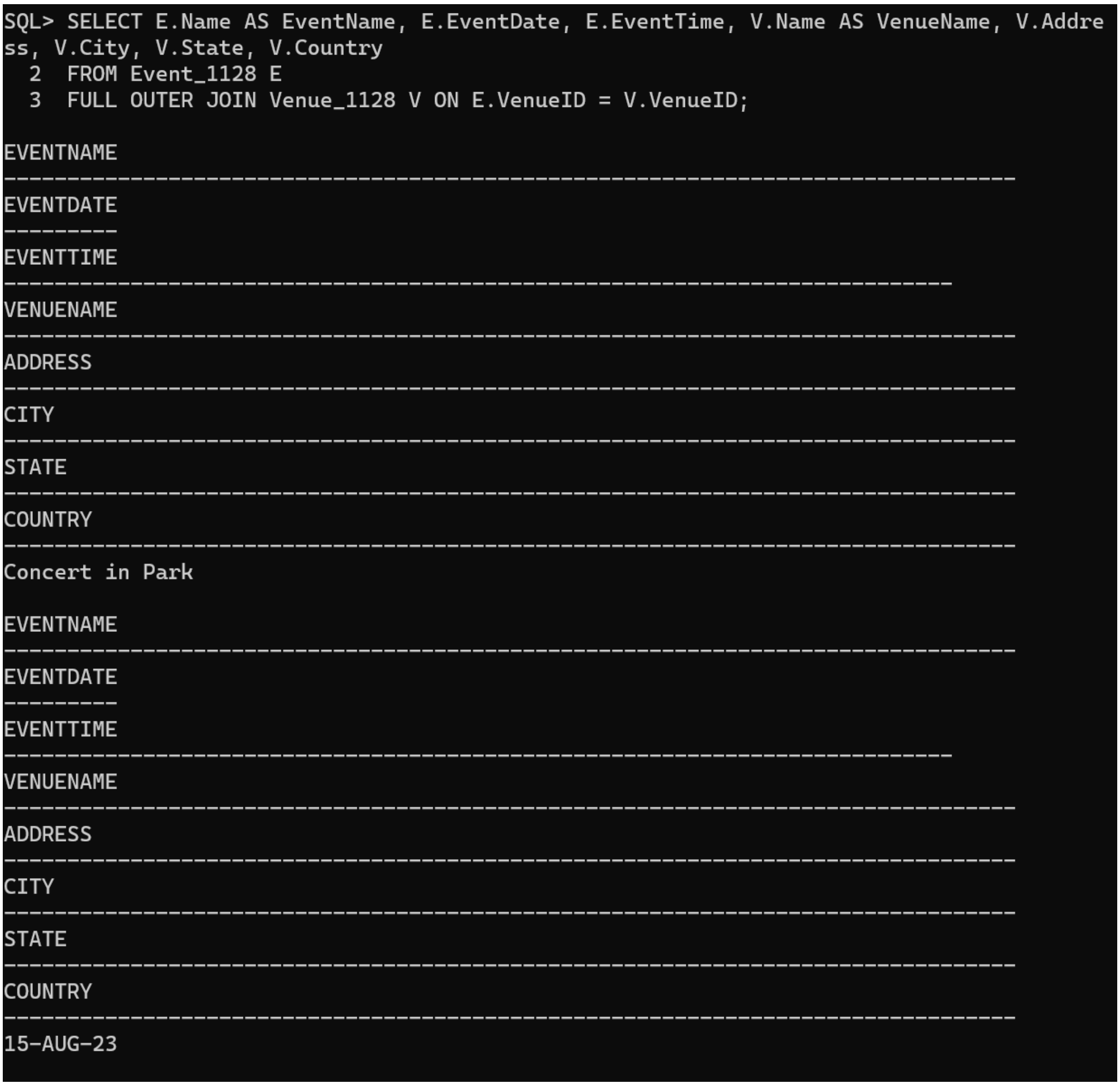
```
SQL> SELECT U.Name
     FROM User_1128 U
     LEFT JOIN Ticket_1128 T ON U.UserID = T.UserID
     WHERE T.TicketID IS NULL;

no rows selected
```

9. Perform Left Outer Join to Retrieve Event Details along with Venue Information.



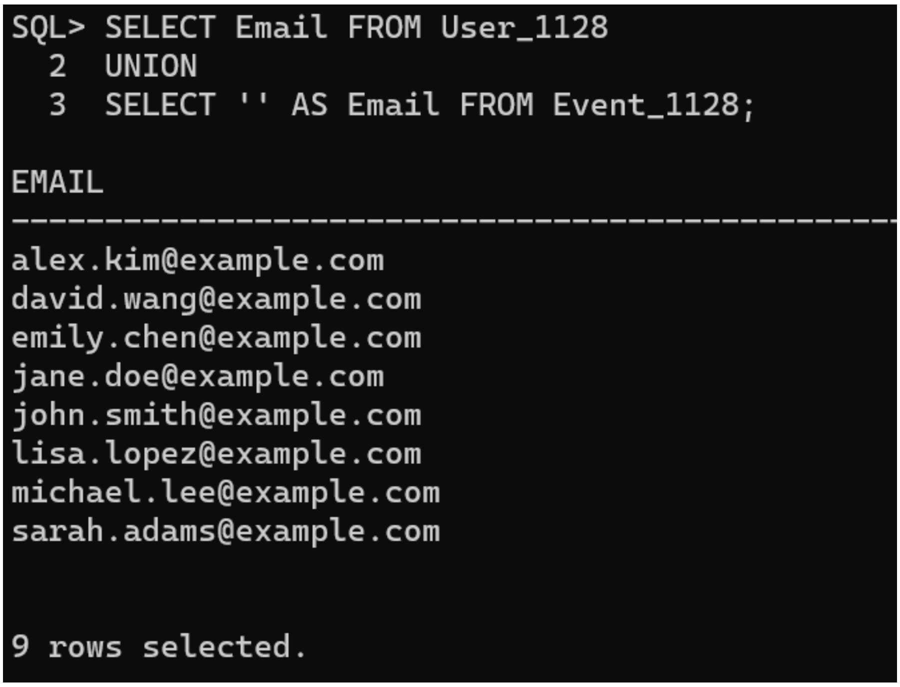
```
SQL> SELECT E.Name AS EventName, E.EventDate, E.EventTime, V.Name AS VenueName, V.Addre
ss, V.City, V.State, V.Country
  2  FROM Event_1128 E
  3  LEFT JOIN Venue_1128 V ON E.VenueID = V.VenueID;

EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
VENUENAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
COUNTRY
--------------------------------------------------------------------------------
Concert in Park

EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
VENUENAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
COUNTRY
--------------------------------------------------------------------------------
15-AUG-23
```

10. Perform Full Outer Join to Retrieve Combined Event and Venue Information.



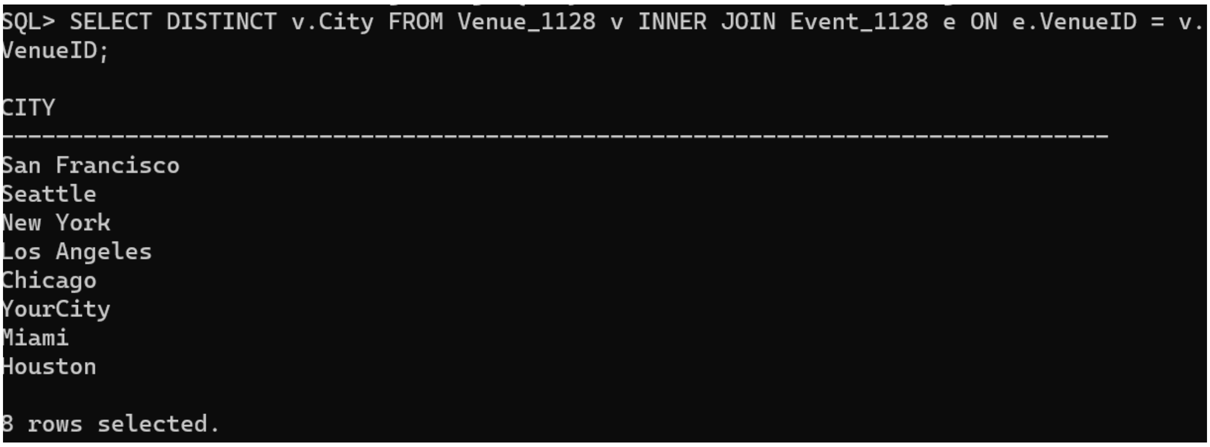
```
SQL> SELECT E.Name AS EventName, E.EventDate, E.EventTime, V.Name AS VenueName, V.Addre
ss, V.City, V.State, V.Country
  2  FROM Event_1128 E
  3  FULL OUTER JOIN Venue_1128 V ON E.VenueID = V.VenueID;

EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
VENUENAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
COUNTRY
--------------------------------------------------------------------------------
Concert in Park

EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
VENUENAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
COUNTRY
--------------------------------------------------------------------------------
15-AUG-23
```

11. Write a query to retrieve a list of unique email addresses from both the User and Event tables using set operations.



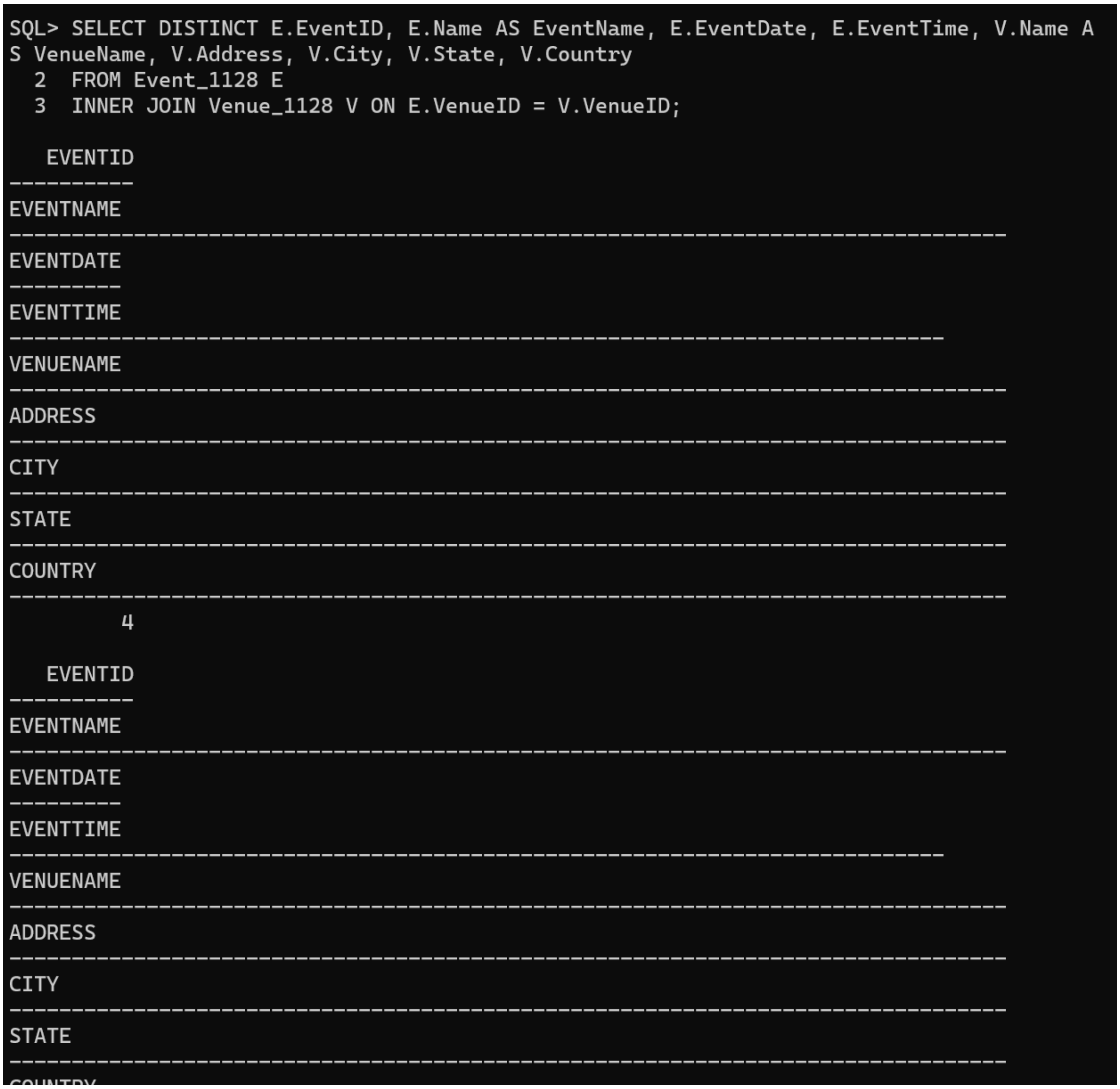
```
SQL> SELECT Email FROM User_1128
  2  UNION
  3  SELECT '' AS Email FROM Event_1128;

EMAIL
------------------------------------------------
alex.kim@example.com
david.wang@example.com
emily.chen@example.com
jane.doe@example.com
john.smith@example.com
lisa.lopez@example.com
michael.lee@example.com
sarah.adams@example.com


9 rows selected.
```

12.Write a query to retrieve a list of unique cities where events are scheduled or venues are located.



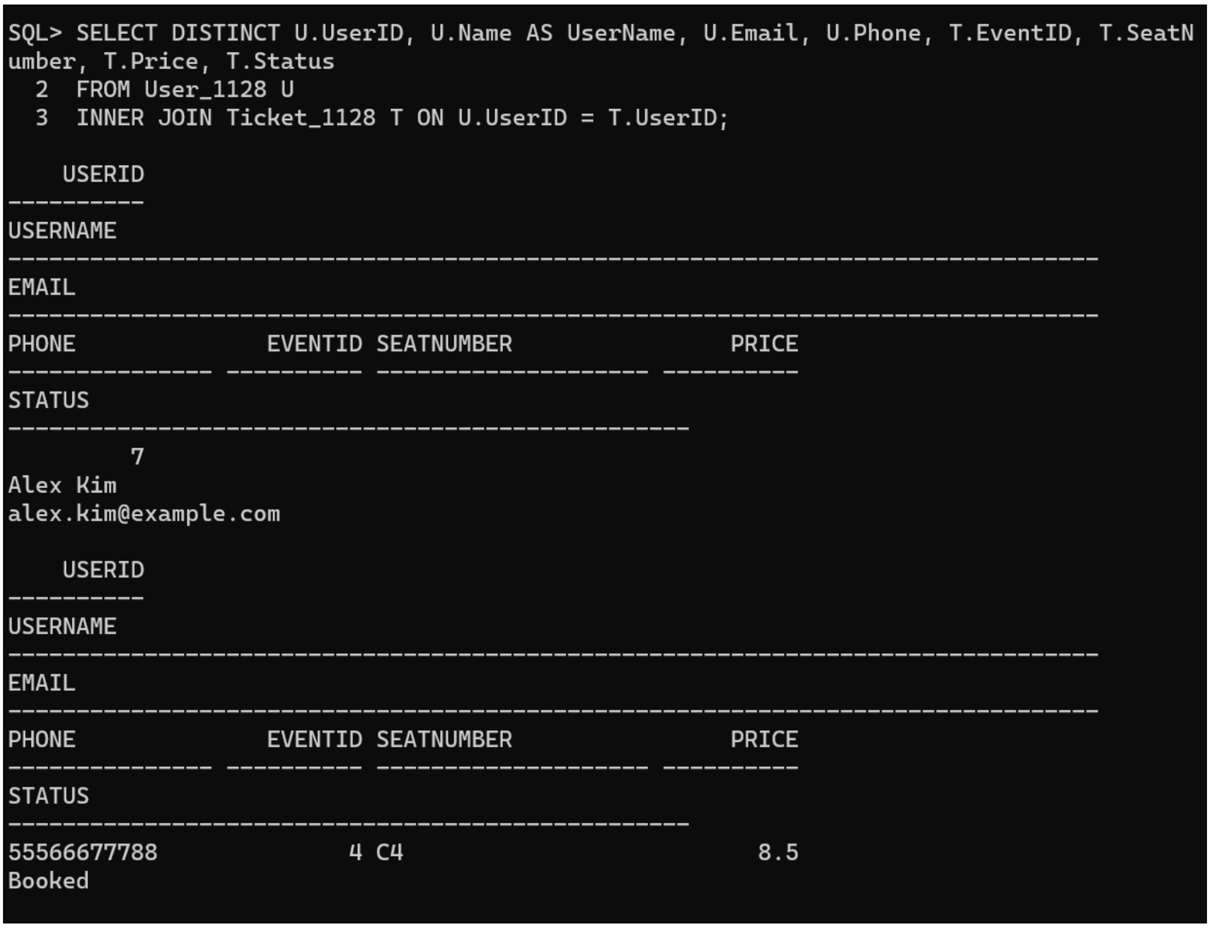
```
SQL> SELECT DISTINCT v.City FROM Venue_1128 v INNER JOIN Event_1128 e ON e.VenueID = v.
VenueID;

CITY
--------------------------------------------------------------------------------
San Francisco
Seattle
New York
Los Angeles
Chicago
YourCity
Miami
Houston

8 rows selected.
```

13.Write a query to display the details of Venue ID conducted in the same Venues.



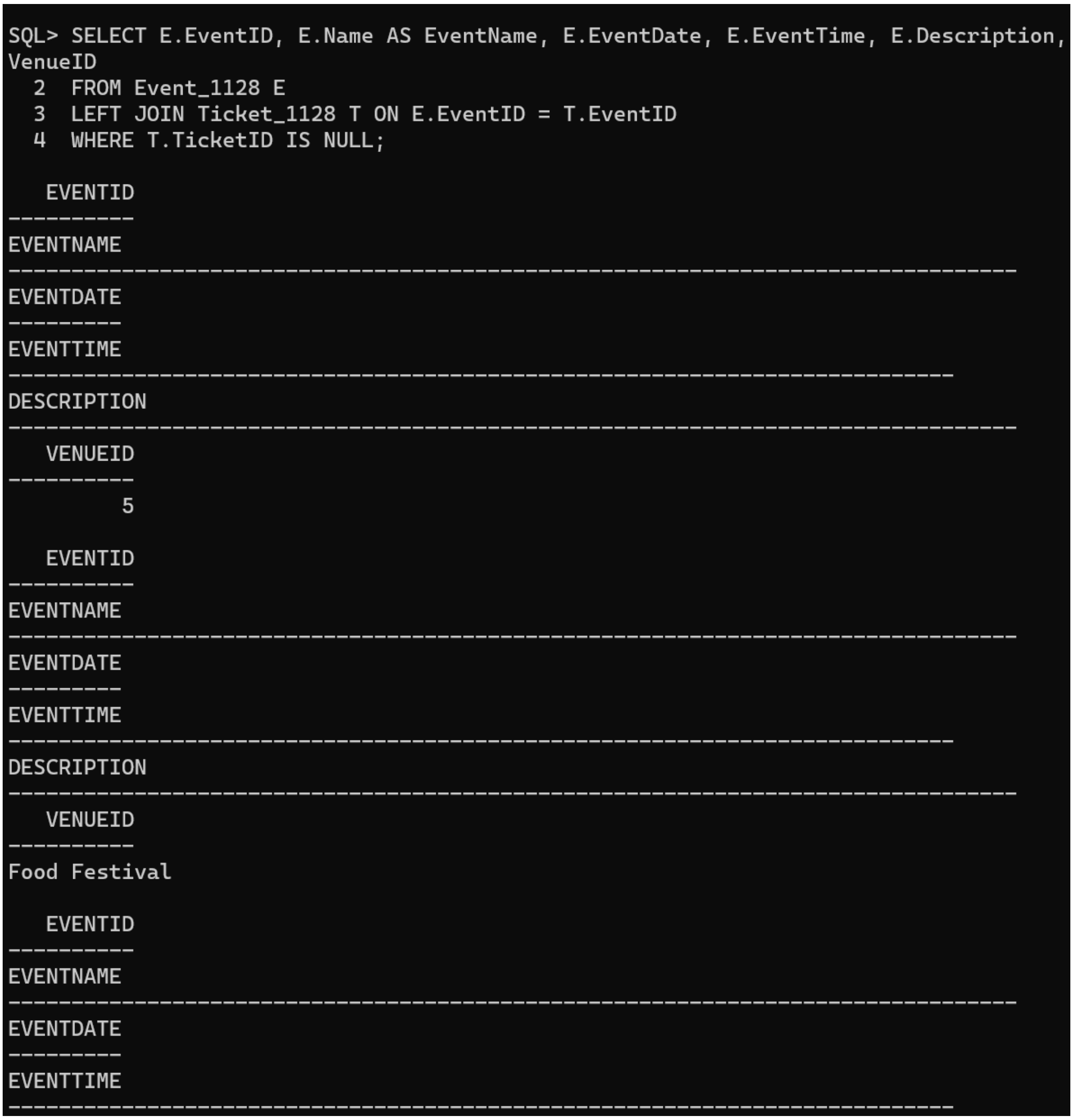
```
SQL> SELECT DISTINCT E.EventID, E.Name AS EventName, E.EventDate, E.EventTime, V.Name A
S VenueName, V.Address, V.City, V.State, V.Country
  2  FROM Event_1128 E
  3  INNER JOIN Venue_1128 V ON E.VenueID = V.VenueID;

   EVENTID
----------
EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
VENUENAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
COUNTRY
--------------------------------------------------------------------------------
         4

   EVENTID
----------
EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
VENUENAME
--------------------------------------------------------------------------------
ADDRESS
--------------------------------------------------------------------------------
CITY
--------------------------------------------------------------------------------
STATE
--------------------------------------------------------------------------------
COUNTRY
```

14. Write a query to display the details of User ID who are users and have registered for an event.



```
SQL> SELECT DISTINCT U.UserID, U.Name AS UserName, U.Email, U.Phone, T.EventID, T.SeatN
umber, T.Price, T.Status
     FROM User_1128 U
     INNER JOIN Ticket_1128 T ON U.UserID = T.UserID;

    USERID
----------
USERNAME
--------------------------------------------------------------------------------
EMAIL
--------------------------------------------------------------------------------
PHONE               EVENTID SEATNUMBER                PRICE
--------------- ---------- -------------------- ----------
STATUS
--------------------------------------------------
         7
Alex Kim
alex.kim@example.com

    USERID
----------
USERNAME
--------------------------------------------------------------------------------
EMAIL
--------------------------------------------------------------------------------
PHONE               EVENTID SEATNUMBER                PRICE
--------------- ---------- -------------------- ----------
STATUS
--------------------------------------------------
55566677788              4 C4                          8.5
Booked
```

15. Write a query to display the details of Event ID which are events but not booked by any one.



```
SQL> SELECT E.EventID, E.Name AS EventName, E.EventDate, E.EventTime, E.Description,
VenueID
  2  FROM Event_1128 E
  3  LEFT JOIN Ticket_1128 T ON E.EventID = T.EventID
  4  WHERE T.TicketID IS NULL;

   EVENTID
----------
EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
   VENUEID
----------
         5

   EVENTID
----------
EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
DESCRIPTION
--------------------------------------------------------------------------------
   VENUEID
----------
Food Festival

   EVENTID
----------
EVENTNAME
--------------------------------------------------------------------------------
EVENTDATE
---------
EVENTTIME
---------------------------------------------------------------------------
```

Result:

The given queries were executed successfully using set operators and joins.